The Earth System Grid Federation : building a Software Framework of Open Source, Modular Components for Analysis of Large, Distributed Scientific Data

ESGF P2P Development Team

Luca Cinquini, Dan Crichton, Chris Mattmann [NASA/JPL]
Gavin Bell, Bob Drach, Charles Doutriaux, Renata McCoy,
Karl Taylor, Dean Williams [LLNL/PCMDI]
Sigurd Christensen , John Harney, Misha Krassovski, Giri Palanisamy,
Galen Shipman, Feiyi Wang, [ORNL]
Ansley Manke, Roland Schweitzer [NOAA/PMEL]
Rachana Ananthakrishnan, Neill Miller [ANL]
Estani Gonzales, Stephan Kindermann [DKRZ]
Philip Kershaw, Stephen Pascoe [BADC]
Sandro Fiore, Giovanni Aloisio [CMCC/University of Salento]
Cecelia DeLuca, Sylvia Murphy, Luca Cinquini [NOAA/ESRL]
...et al...

ESGF
Earth System Grid Federation

JPL
Jet Propulsion Laboratory
California Institute of Technology

ESGF Overview

The Earth System Grid Federation (ESGF) is a spontaneous, unfunded collaboration of people and institutions working together to build an open source software infrastructure for the management and analysis of Earth Science data

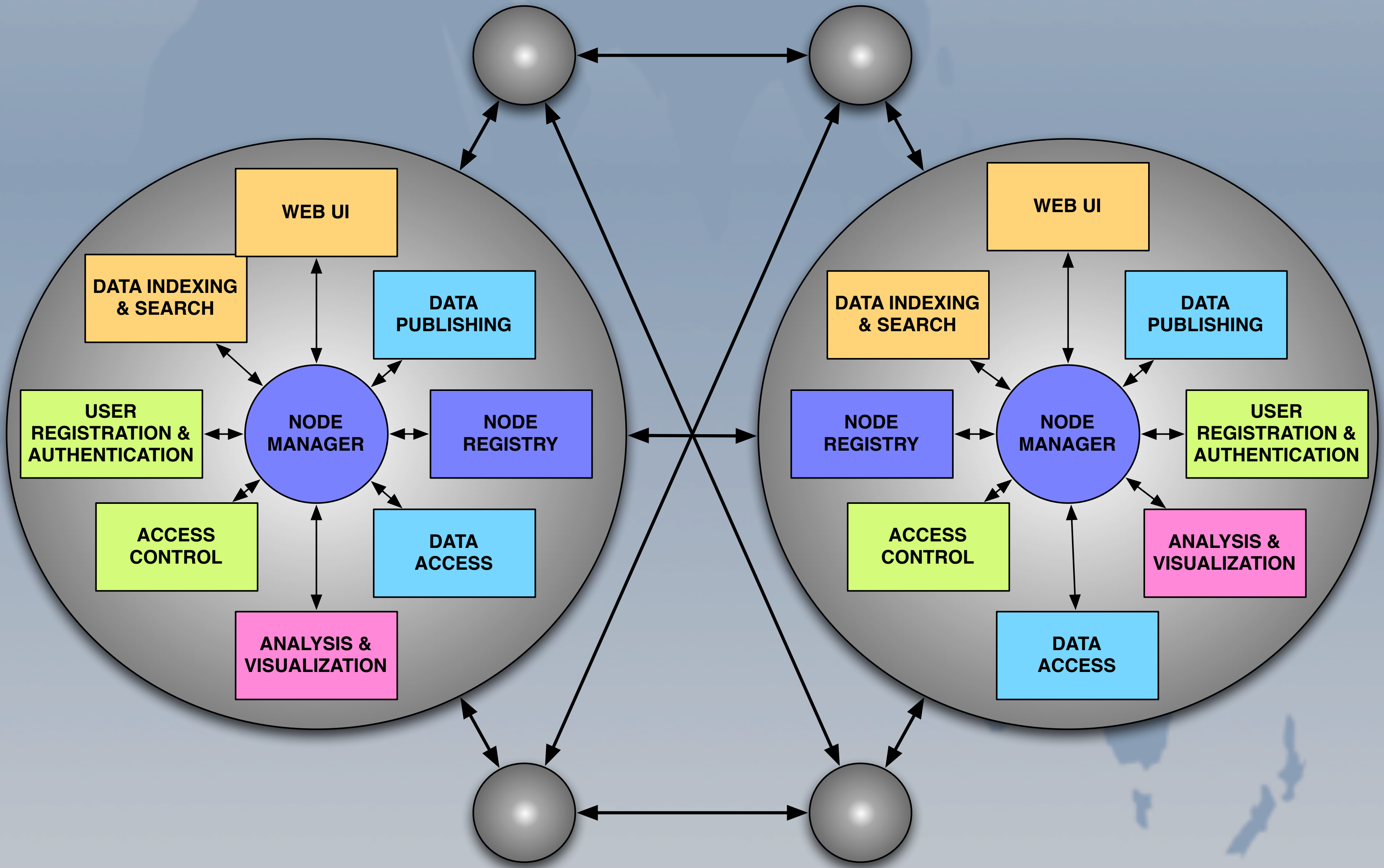
- First meeting @ GFDL in July 2010: BADC, DKRZ, PCMDI, ORNL, NCAR, GFDL, JPL
- Security meeting @ Argonne in September 2010: ANL, BADC, NCAR, PCMDI, JPL
- Support current CMIP5/IPCC-AR5 activities, and prepare for future IPCC assessments
- Develop data and metadata facilities for inclusion of observations in CMIP5
- Evolve current ESG architecture through sound software principles:
  - Modularity: applications can be developed, deployed, configured independently
  - Code to interfaces and APIs so implementation can be swapped, evolved
  - Use standards and profiles to achieve interoperability
- Open Source license (BSD), freely available
  - git clone git@esg-repo.llnl.gov:<module name>.git
- Collaborate across agencies and political boundaries
- Integrate/interoperate with other existing infrastructures: NASA, NOAA, ESIP, ES-INES, ...
- Open development: joint ownership, prioritization, responsibility
- Multi-language environment: Java and Python

The Peer-To-Peer ESGF Node

Peer-To-Peer architecture (p2p): system of distributed nodes that have in principle the same capabilities, interact on an equal base, can share resources, can act as producers or consumers of services. No central coordination, no single point of failure, no rigid distinction of roles as in a traditional client-server architecture.

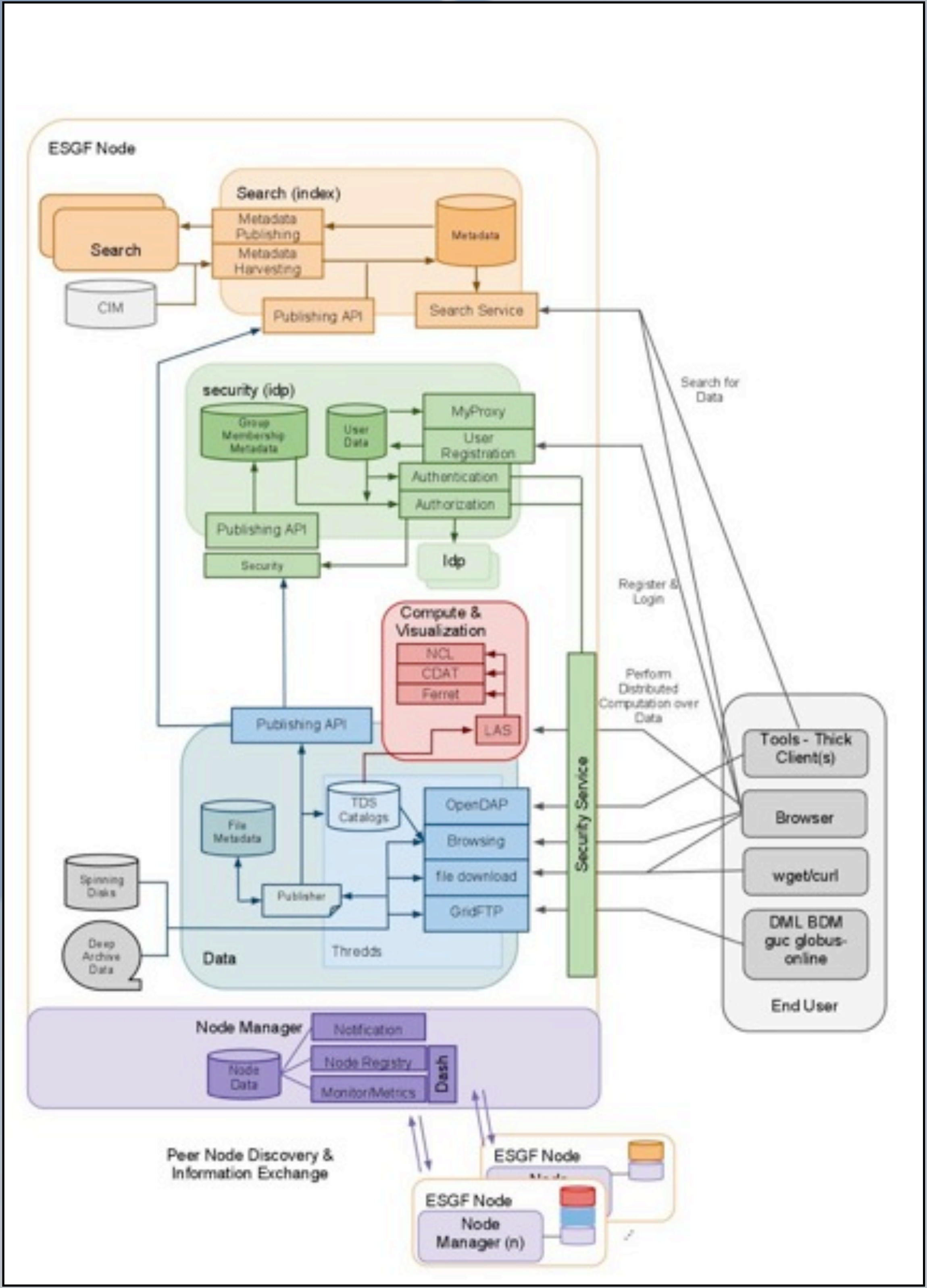
Features of ESGF p2p Node:

- Expanded functionality
- Configurability
  - Data, Index, Compute, IdP
- Services can be added more easily by each institution
- Resources can be scaled as necessary by deploying more Nodes of appropriate type
- Elasticity: Nodes can join/ leave federation dynamically



Summary of Recent ESGF Development

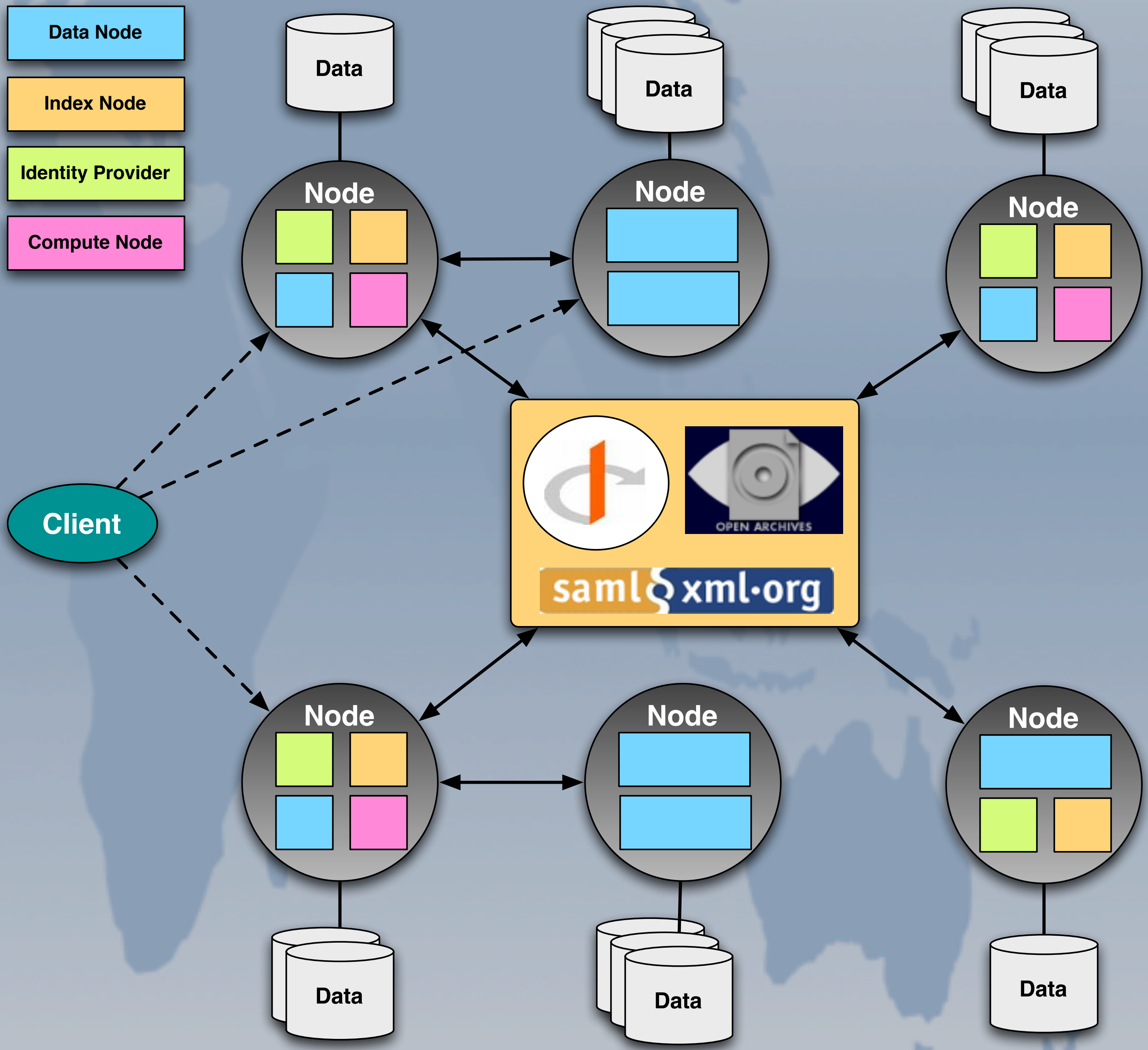
- Establishment of metadata conventions for observations
- More modular security infrastructure, with expanded functionality
- New back-end search services based on Solr
- New web front end application
- Live Access Server integration
- Expanded configurability for Node installation
- ESGF Registry Service
- Dashboard application to monitor p2p system
- Rich client access to ESGF services
  - OpenDAP access w/ X509 certificates
  - UV-CDAT querying of Search Services
- Preliminary integration with Apache/NASA OODT stack for data management
- Deployment of support infrastructure @ PCMDI: GIT repo, Bugzilla, wiki, Artifactory...



ESGF Security Services: Overview

ESGF is composed of centers that are autonomously managed: each center registers its own users, and establishes access control over its resources. Seamless data access within the federation is made possible by adoption of a federated security model:

- Single Sign On: users may register at any site, then authenticate to any other site
- Distributed Access Control: resources can be controlled by local policies, or policies that are managed by another site (“CMIP5”)



ESGF Security Services: Overview

- ESGF security infrastructure is based on standards:
  - SSL: encryption of information for internet communication
  - OpenID: SSO protocol for browser based access
  - PKI/X509: credential management for rich client based access
  - SAML: XML for encoding and signing authentication/authorization assertions
- ESGF security specifications implemented in Java and Python
  - A python client can access resources behind a Java server, and vice-versa
  - Can secure any Java or Python web application
- Applies to data access via HTTP, OpenDAP and GridFTP
- Non-intrusive: security applied outside of application itself
  - Server side: front-end filters that intercept request before application
  - Client side: libraries that transmit certificate as part of the request
- Federation Registry stores and broadcasts information about security service endpoints, site trust roots, white lists etc.
- Collaborative work of many institutions: ANL, BADC, JPL, PCMDI, DKRZ, PMEL
- Code is open source (BSD license) stored in PCMDI GIT repositories

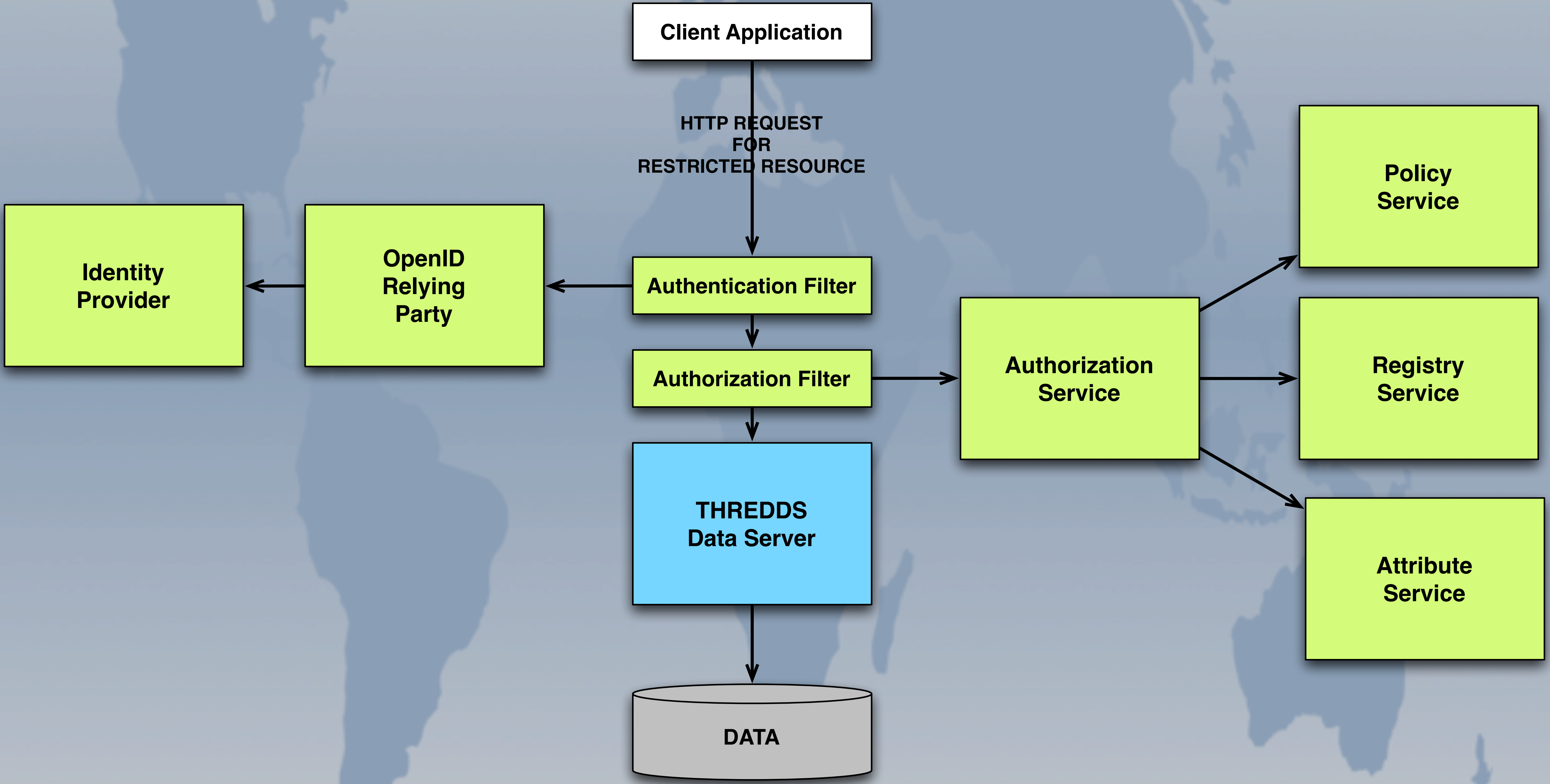
ESGF Security Services: Java Components

Java implementation of ESGF security services is a modular framework of web applications that can be run independently or in concert:

- Identity Provider: user authentication via OpenID, pluggable back-end
- Attribute Service: SAML assertions about user information and access control attributes
- Authorization Service: SAML authorization assertions (wether a user can execute a given operation on a set of resources)
- OpenID Relying Party: consumes authentication assertions as X509 certificates or OpenID statements from an IdP, and caches it for the working session
    - New: uses white list from ESGF Registry Service
- Security Filters: extensible framework of servlet filters that may be used to secure any Java servlet application
    - New: authorization filter can query multiple authorization services

All components are coded to Java interfaces > allow multiple implementations and back-ends

ESGF Security Services: Example Use Case

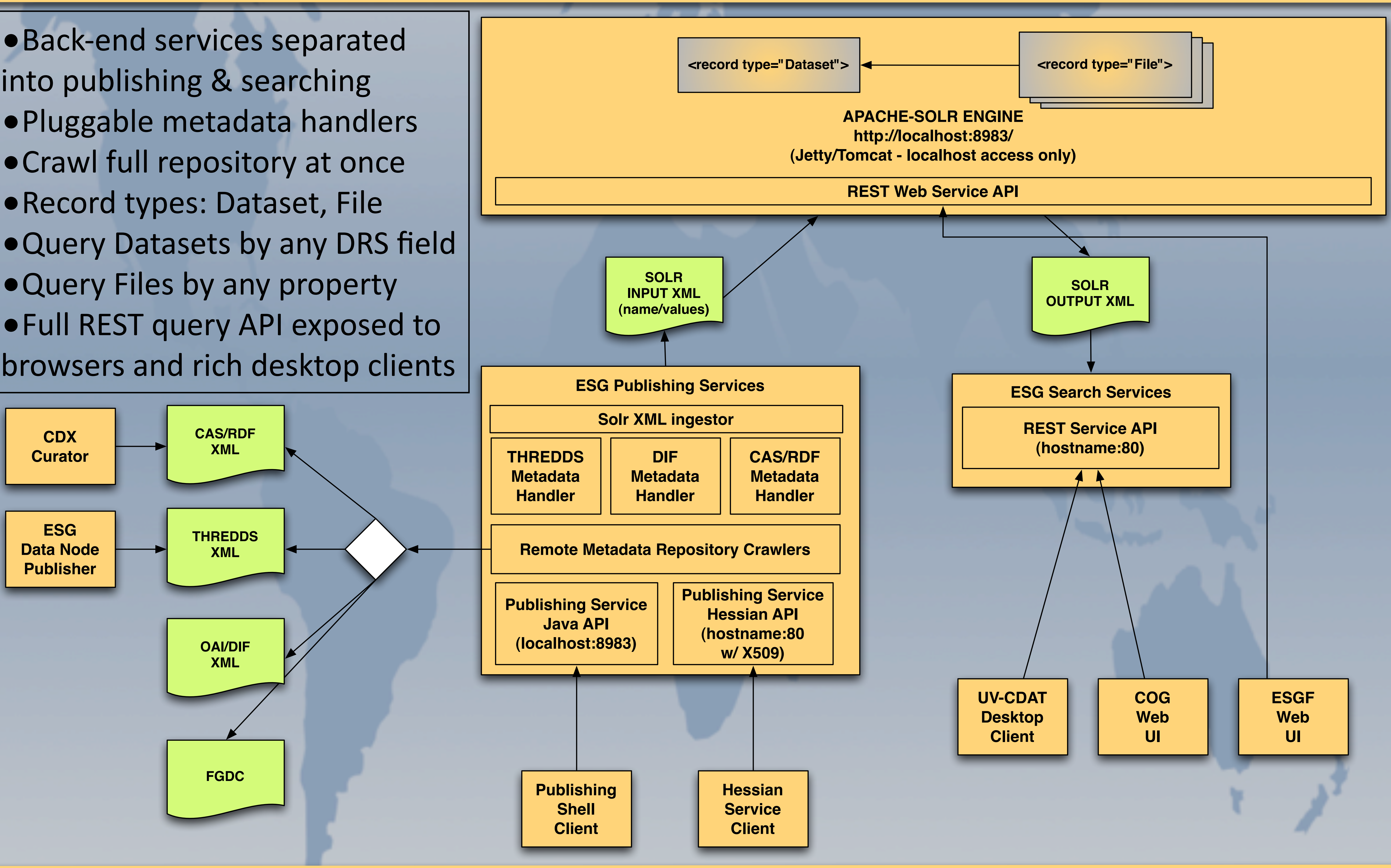


ESGF Search Services: Overview

- ESGF has developed a new metadata search capability based on Apache Solr
  - Powerful text search based on Lucene
  - High performance, proven scalability
  - Faceted search including facet counts
  - Scoring, stop words,...
  - Flat (key, value) pair model
- Rationale:
  - More extensible infrastructure for inclusion of observations metadata
    - Easy to add new facets: info flows automatically from files to Solr
    - Geospatial and temporal search
  - Support ingestion of metadata from different sources
  - Support rich desktop clients through powerful and extensible REST API
- Back-end services for indexing, querying developed independently from UI
  - Multiple UIs querying Solr services, same UI querying different back-ends
- Publishing services exposed through ESG Hessian API
  - ESG publisher can be used to publish metadata into a p2p ESGF Node

ESGF Search Services: Architecture

- Back-end services separated into publishing & searching
- Pluggable metadata handlers
- Crawl full repository at once
- Record types: Dataset, File
- Query Datasets by any DRS field
- Query Files by any property
- Full REST query API exposed to browsers and rich desktop clients



<record type="Dataset">

<record type="File">

APACHE-SOLR ENGINE
http://localhost:8983/
(Jetty/Tomcat - localhost access only)

REST Web Service API

SOLR INPUT XML (name/values)

SOLR OUTPUT XML

ESG Publishing Services

Solr XML ingestor

THREDDS Metadata Handler

DIF Metadata Handler

CAS/RDF Metadata Handler

Remote Metadata Repository Crawlers

Publishing Service Java API (localhost:8983)

Publishing Service Hessian API (hostname:80 w/ X509)

ESG Search Services

REST Service API (hostname:80)

CDX Curator

CAS/RDF XML

ESG Data Node Publisher

THREDDS XML

OAI/DIF XML

FGDC

Publishing Shell Client

Hessian Service Client

UV-CDAT Desktop Client

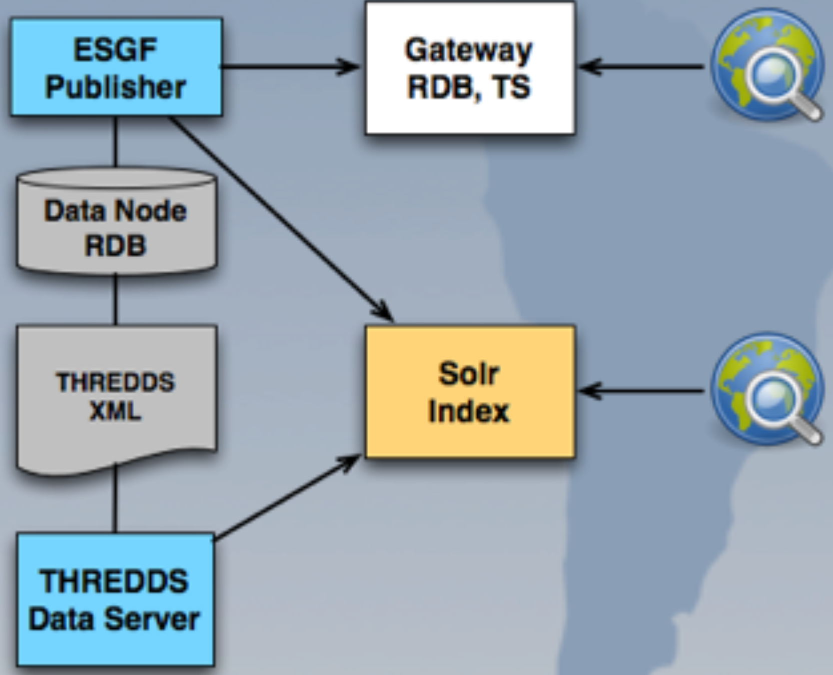
COG Web UI

ESGF Web UI

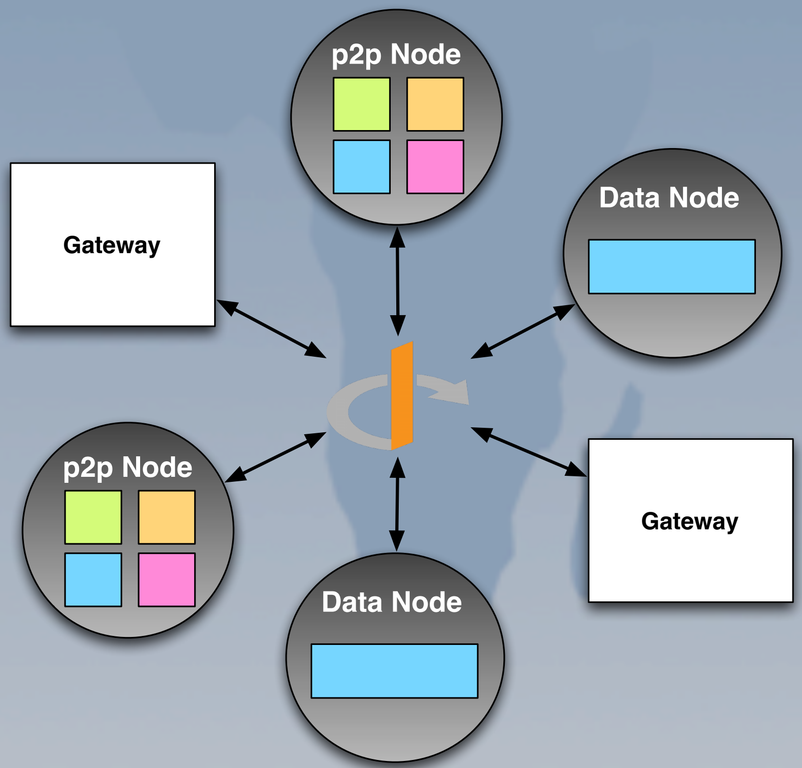
Interoperability

A p2p Node can be deployed within the existing Earth System Grid Federation to work inter-operably with Data Nodes and Gateways

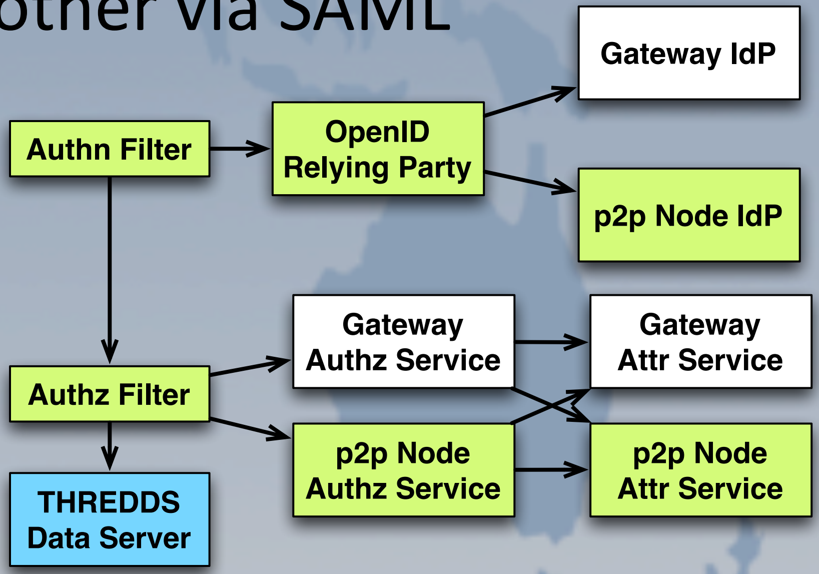
- Data can be published to a Gateway and a p2p Node...
- ...or harvested from top-level THREDDS catalog



- Users can register at a p2p Node or a Gateway and SSO throughout the federation

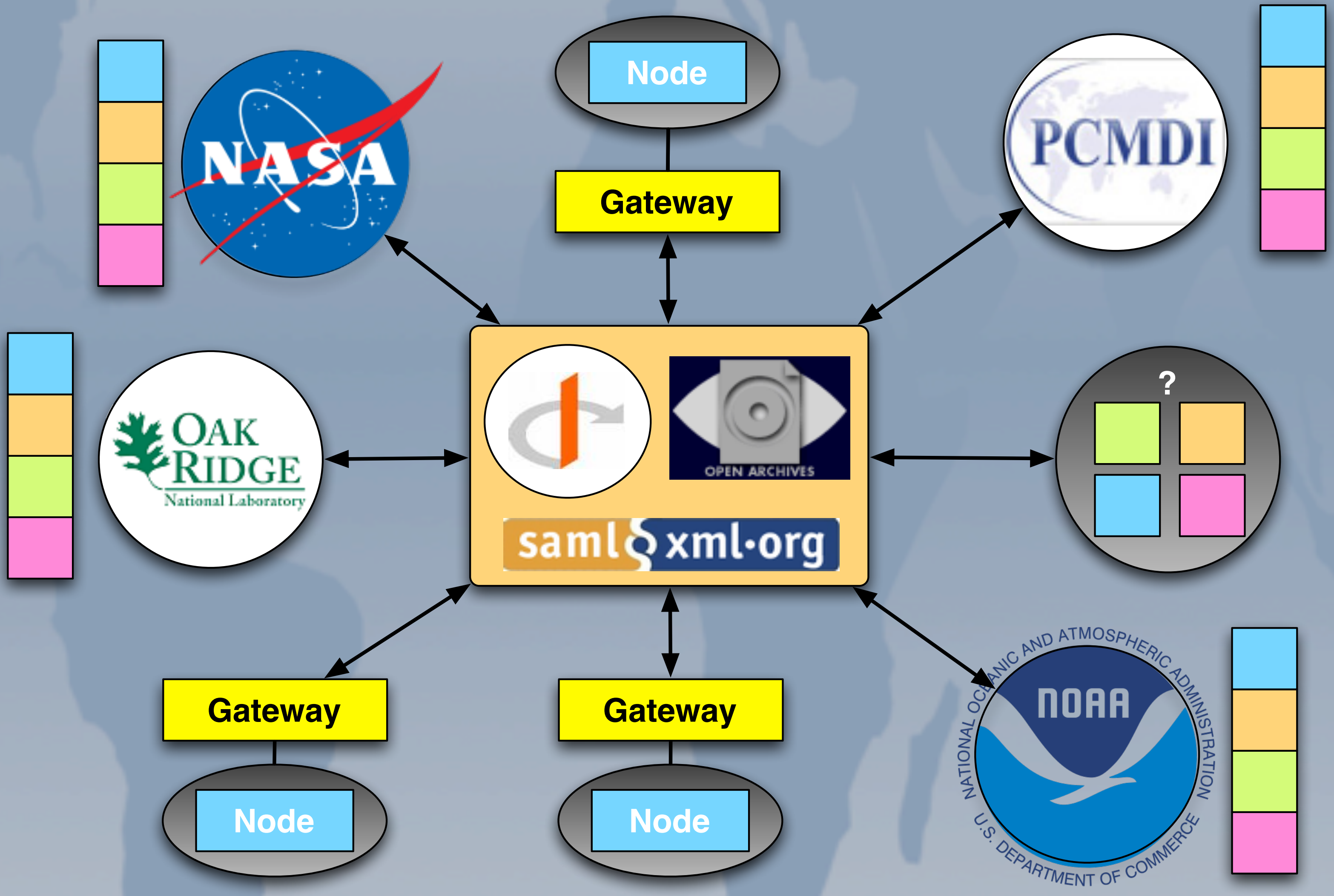


- Users can request data at a Data Node and be authenticated/authorized by services at Gateway or p2p Node
- Services can query each other via SAML



ESGF p2p Testbed

Setting up ESGF peer-to-peer testbed to include observations in CMIP5 @ NASA JPL, ORNL, PCMDI, NOAA ESRL, others?


Node
Gateway
NASA
PCMDI
OAK RIDGE National Laboratory
OPEN ARCHIVES
saml xml·org
?
Gateway
Gateway
Node
Node
NOAA
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
U.S. DEPARTMENT OF COMMERCE

We welcome closer collaboration and joint ownership with everybody, under an open source model for collaborative development and governance.

Questions ?

<u>Upcoming Presentations on ESGF P2P Architecture:</u>

- Gavin Bell (PCMDI):
  - *"ESGF: How to build an elastic distributed system over 'Big Data'"*
- Feiyi Wang, Galen Shipman, and John Harney (ORNL):
  - *"The Earth System Grid Federation User Interface"*